ML APPLICATIONS AT THE ADVANCED PHOTON SOURCE

NIKITA KUKLEV (ADVANCED PHOTON SOURCE, ANL, USA)
LOUIS EMERY
HAIRONG SHANG
MICHAEL BORLAND
YINE SUN

Optics Tuning and Corrections for Future colliders workshop
June 26, 2023

APS UPGRADE

APS is undergoing a major upgrade (APS-U)

- New storage ring
- Refurbishment of injector complex
- More beamlines

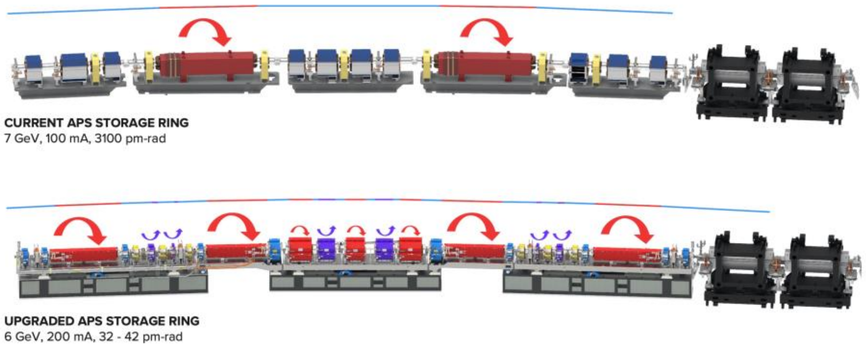
Extremely compressed timeline

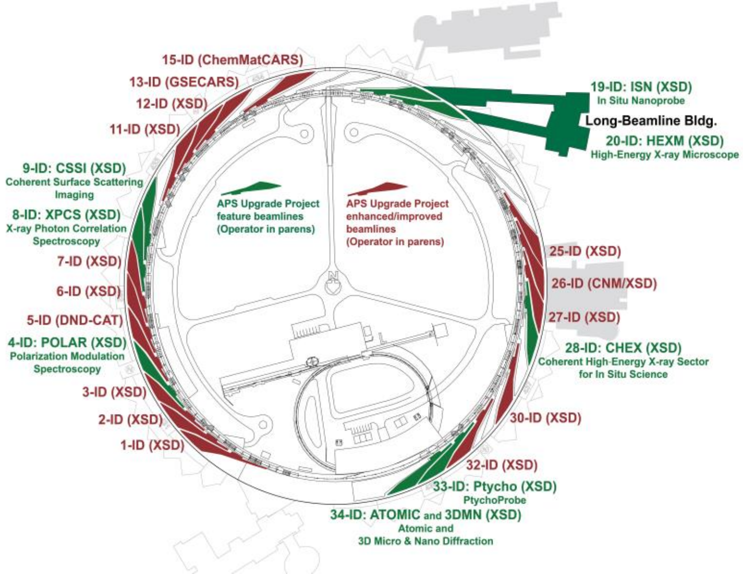
- **<3 months** for 24x7 beam commissioning

New opportunities and new problems

- More challenging physics
- New diagnostics and increased data rates

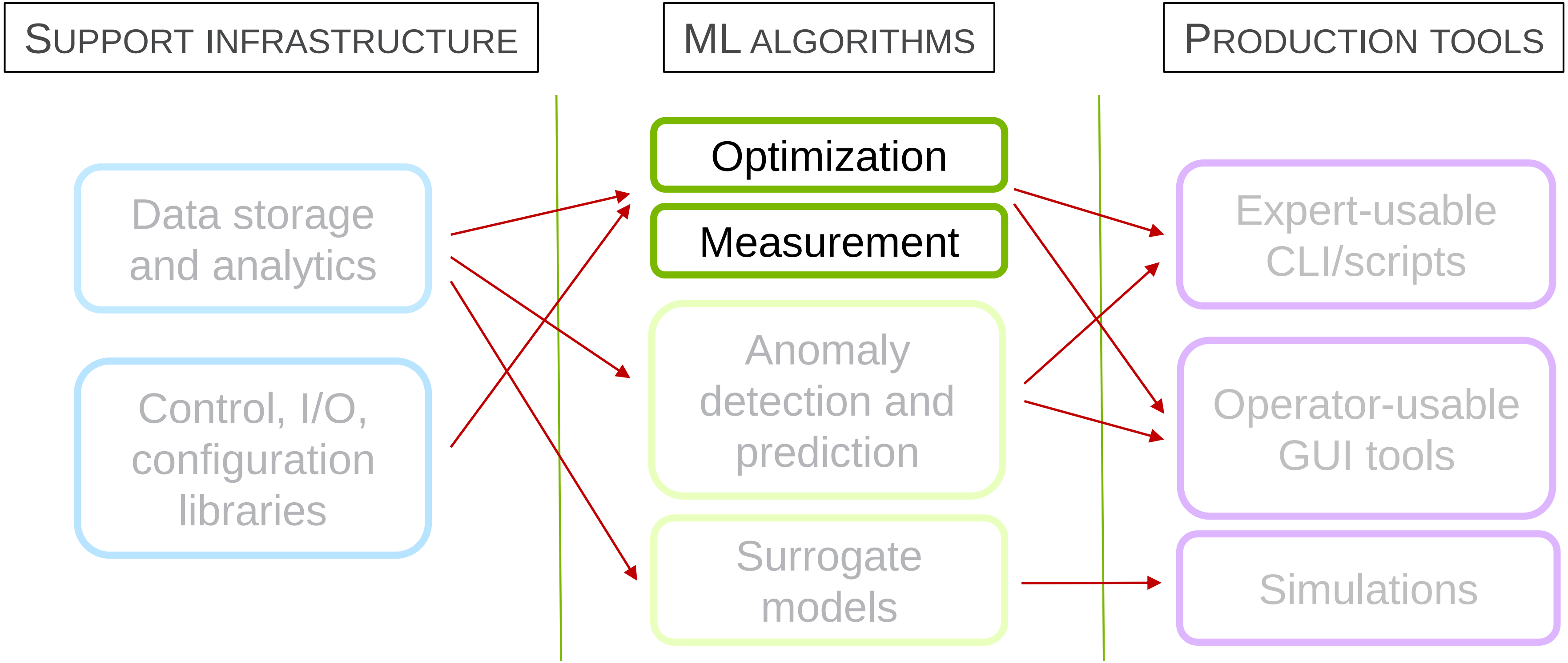
See talk by V. Sajaev for details of commissioning





ML @ APS

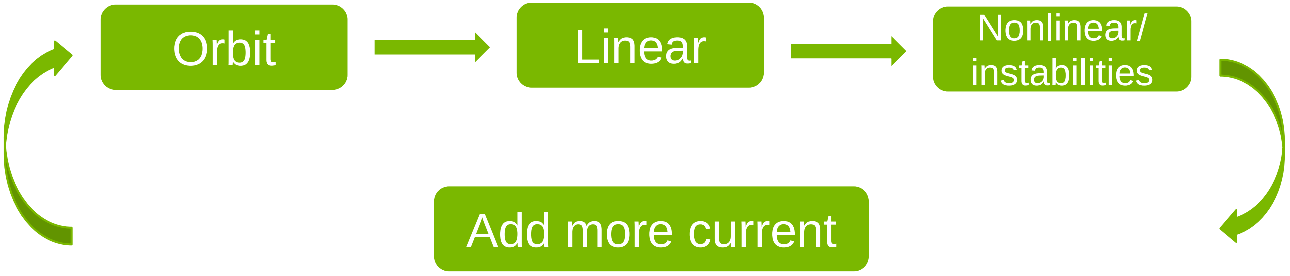
Several efforts underway aimed at APS-U commissioning and operations



N. Kuklev et al., IPAC23 (WEPL004, THPL007, THPL008)
I. Lobach et al., IPAC23 (THPL005)

ML @ APS - OPTIMIZATION

Large part of commissioning process is spent iteratively tuning linear/nonlinear optics



Many talks at this workshop discuss details of such strategies

However, eventually might get stuck:

- Model does not match reality
- Unexpected physics
- etc.

(NON-PARAMETRIC) ACCELERATOR OPTIMIZATION

Solution – switch to non-parametric methods and start tweaking inputs. Many methods:

Humans

Classic methods

Artificial Intelligence

Machine Learning

Deep Learning

Scientists
Engineers
Operators

Nelder-Mead Simplex
RCDS
Extremum Seeking

Genetic Algorithm
Particle Swarm

Bayesian Optimization

Reinforcement Learning
Deep Kernel Learning

Adaptive (time-aware) BO to compensate drifts

High-dimensional single/multi-objective BO for large parameters spaces

Work @ APS

WHAT IS MO OPTIMIZATION USED FOR?

Light sources designs are optimized heavily:

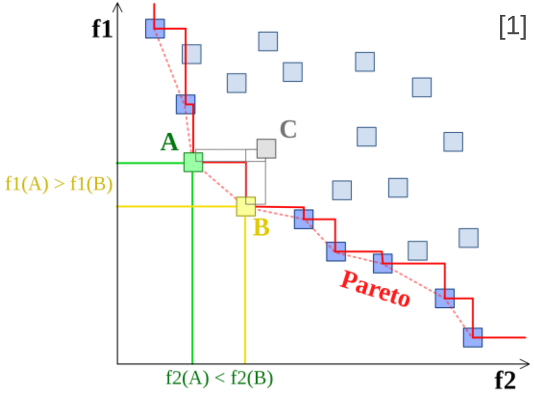
- Dynamic acceptance (DA)
- Lifetime (momentum aperture, MA)
- Detuning, off-momentum DA, CS, etc.

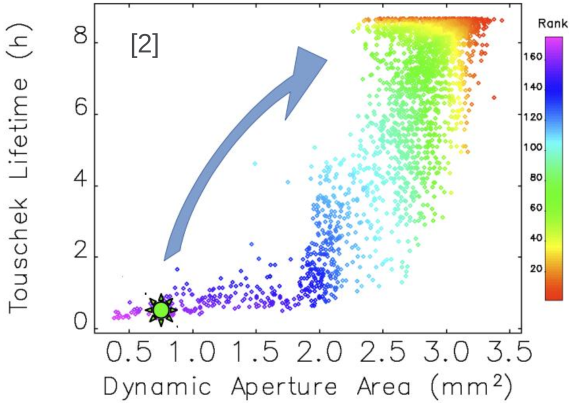
Online use demonstrated previously with:

- MOGA
- MOPSO
- MGGPO

Significantly harder than single objective, required **lots** of beam time

How to improve?





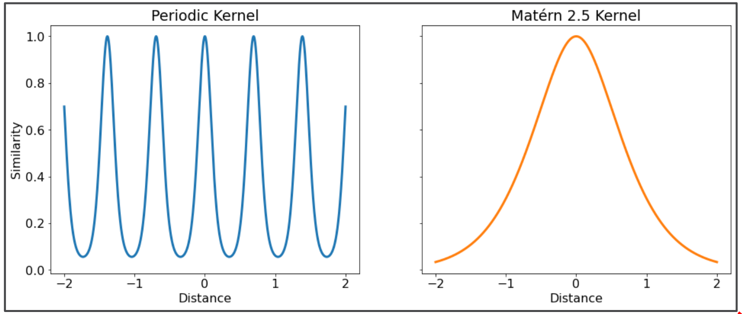
1. https://en.wikipedia.org/wiki/Pareto_front#/media/File:Front_pareto.svg
2. M. Borland et al., HEP GARD Accelerator and Beam Physics: Workshop #2, WG 4 (2020)
3. M. Borland et al., J. Synchrotron Rad. 21 (2014).
4. Y. Sun, NAPAC2016 (WEPOB15)
5. Y. Sun, NAPAC2016 (WEPOB12)

BAYESIAN OPTIMIZATION

INPUTS

$\mathbf{x}$

MACHINE BLACK BOX



$$\mathbf{y} = f(\mathbf{x}) + \varepsilon$$

Outputs

$$f(\mathbf{x}) \sim \mathcal{GP}(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}'))$$

GAUSSIAN PROCESS (GP) MODEL

$$\mathrm{p}(\mathbf{f} \mid \mathbf{y}, \mathbf{x})$$

$$\mathrm{UCB}(\mathbf{x}) = \mu(\mathbf{x}) + \sqrt{\beta} * \sigma(\mathbf{x})$$

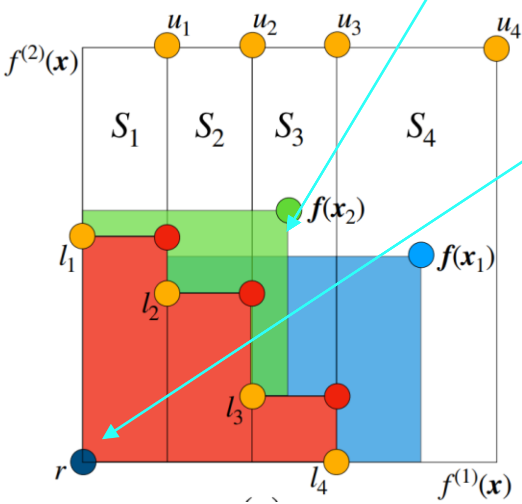
MAXIMIZE ACQUISITION FUNCTION

arXiv:2006.05078
arXiv:2105.08195

BoTorch

MOBO INTERNALS

- Fit Gaussian process model as usual, for each objective separately
- Acquisition function - seek to maximize **hypervolume** relative to **reference point**



- Recent progress in ML frameworks has made this computation **tractable** for online optimization in high dimensional spaces (~seconds)
- (ask me for details)

MOBO DEVELOPMENT @ APS

Want a **sample-efficient** method for **online** nonlinear optics optimization

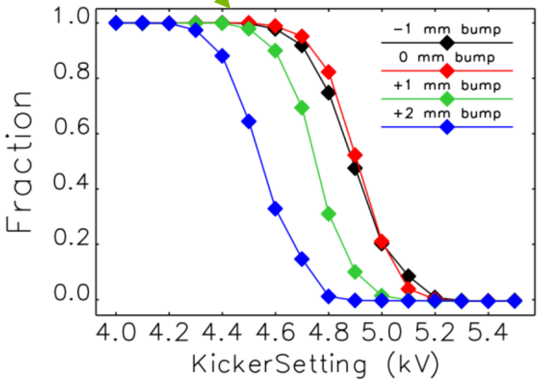
Sample-efficiency has **two** parts:

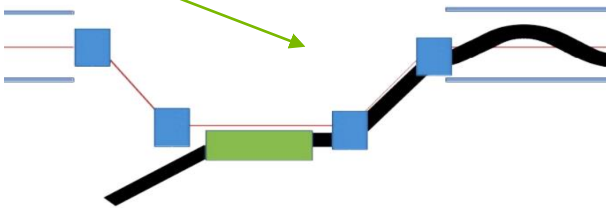
- Time per sample
- ‘Intelligence’ of picking good candidates

Decided to focus on DA/MA since a very well studied set of objectives

PREVIOUS DA/MA MEASUREMENT IN APS

- APS does not have single-bunch kickers (APS-U will)
- Previous efforts measured DA and MA **separately**
  - DA = fill+kick scan (slow) or injection efficiency (with detuned kickers to put beam on DA edge, less robust)



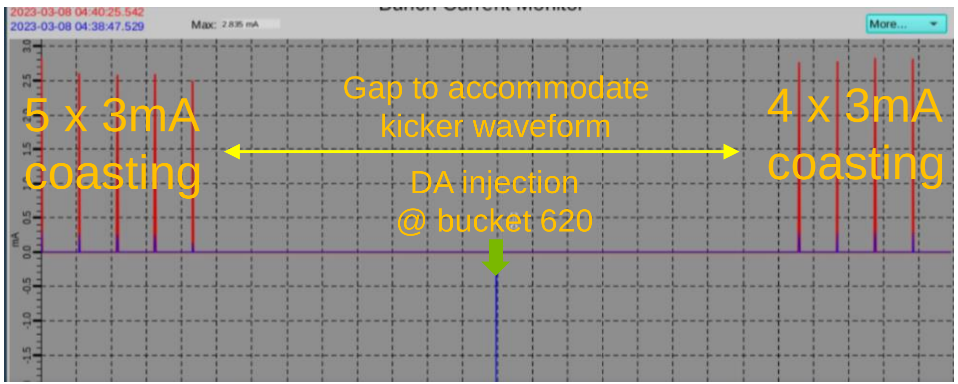


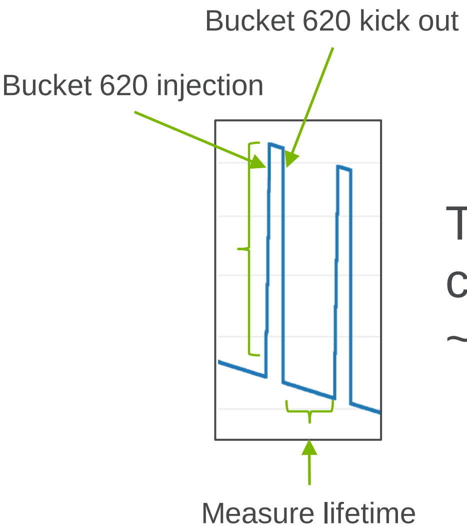
  - MA = lifetime of coasting beam

L. Emery, PRAB 24 (2021)

SPEEDING UP MEASUREMENTS

- Developed new combined procedure based on special bunch pattern
  - Injection efficiency with partial kicker strength, then kick out to clear gap
  - Lifetime measured in coasting bunches that are **not lost during injection**
  - Similar procedure will be used in APS-U, so was good exercise

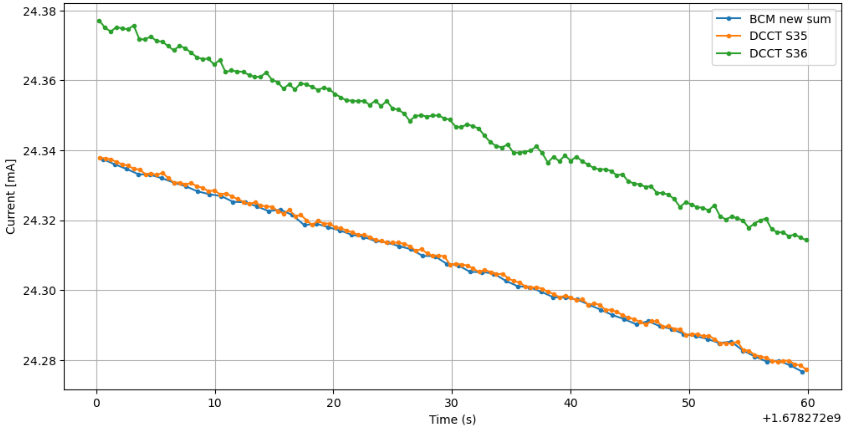


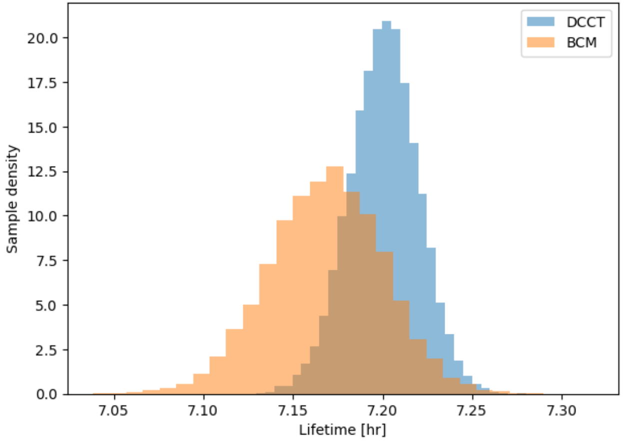


Total cycle ~50s

MEASUREMENT PRECISION

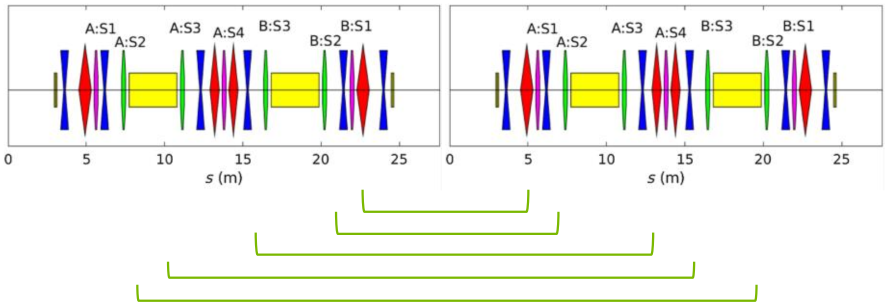
- Both lifetime and injection efficiency measurements depend on DCCT and/or BCM
  - +-10uA DCCT
  - +-100uA BCM per bucket, but sum all buckets
- Need to normalize ($I^{2/3} = I^{1}$ from Touschek x $I^{-1/3}$ from potential well distortion)
- Bootstrapping analysis used to provide lifetime fit parameter distribution
  - Demonstrates that **30s collection is sufficient for +-0.05hr**





EXPERIMENTAL KNOBS

- Instead of using sextupoles directly, work in chroma null space of a 2-cell supersector
- 7D – 2 = 5D
- Prevents algorithm from causing instabilities



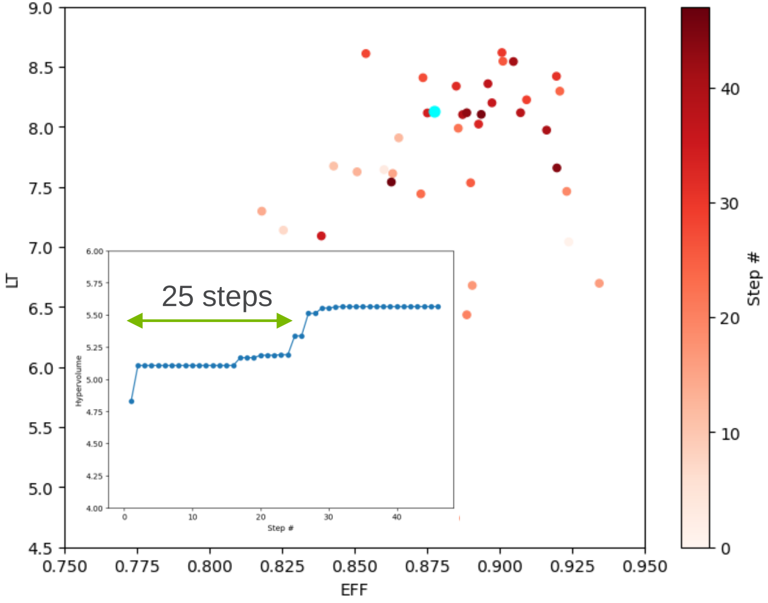
U.S. DEPARTMENT OF ENERGY
Argonne National Laboratory is a U.S. Department of Energy laboratory managed by UChicago Argonne, LLC.

Argonne NATIONAL LABORATORY

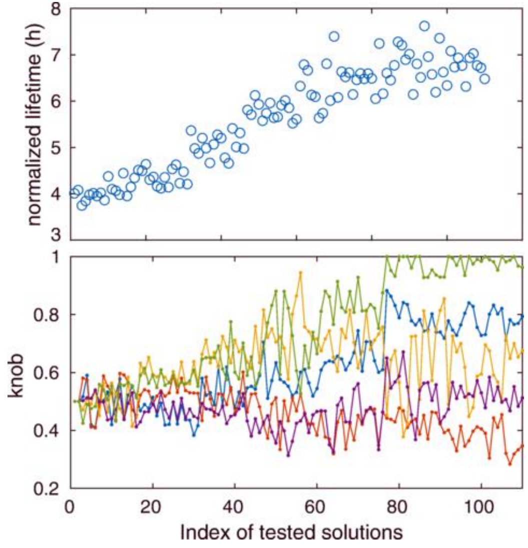
RESULTS – 5D

- Compare with MGGPO results – over 3x efficiency improvement
  - (note: used different lattices, objective values not fully comparable)

MOBO



MGGPO

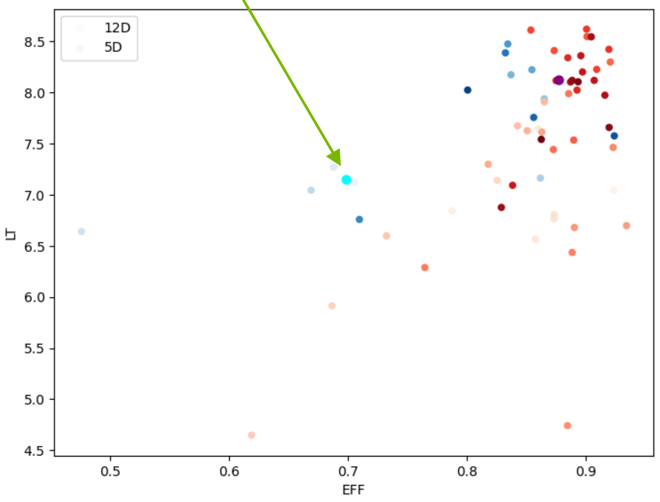


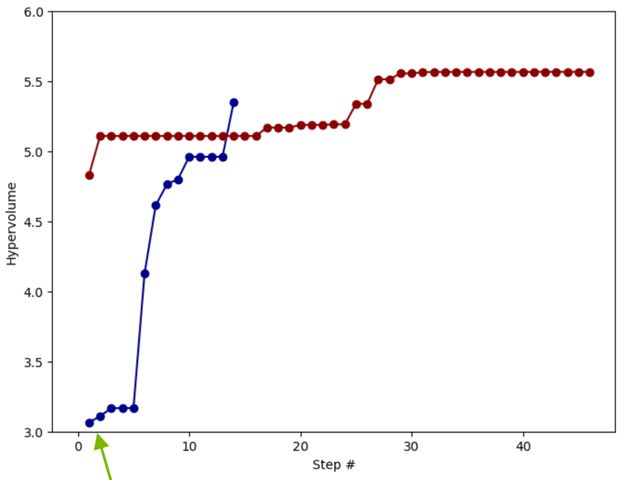
L. Emery, PRAB 24 (2021)

RESULTS – 5D

- Spoiled initial conditions, performance was quickly reattained

New start

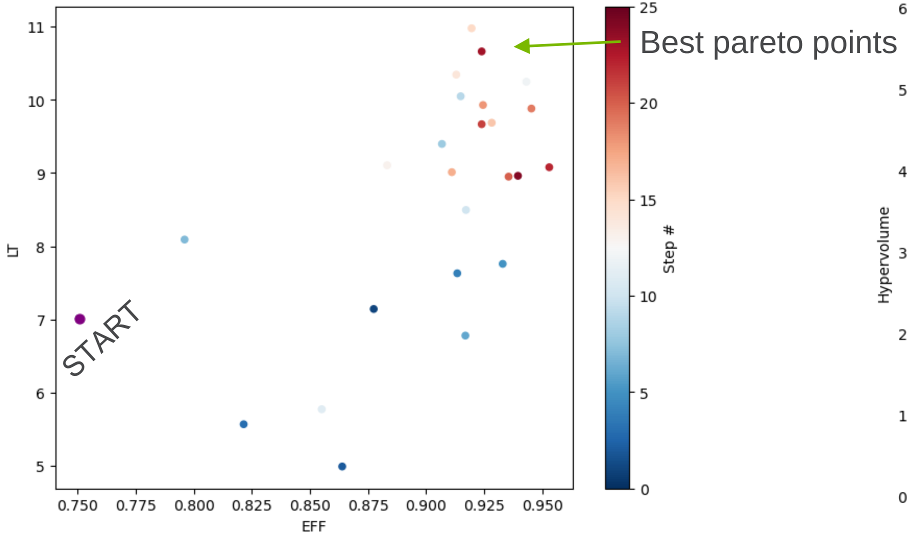




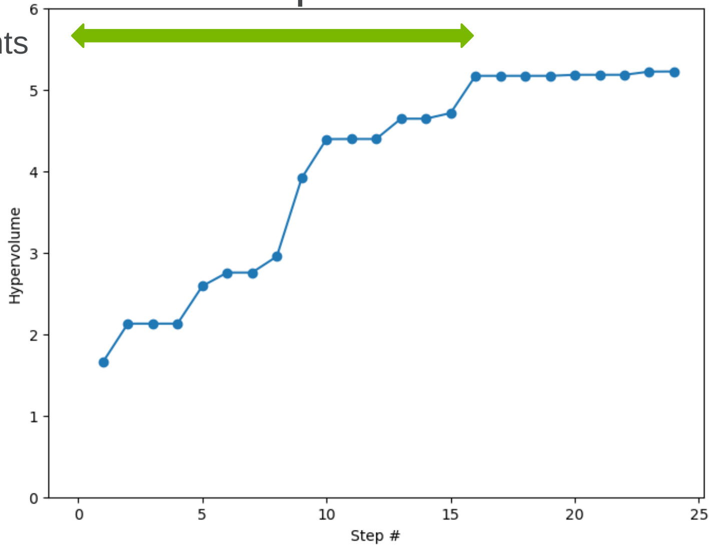
Note start != 0, reference point below initial value

RESULTS – 7D

- Tried on raw sextupoles - performance continued to be very good
- Only small chromaticity changes, had to carefully define limits

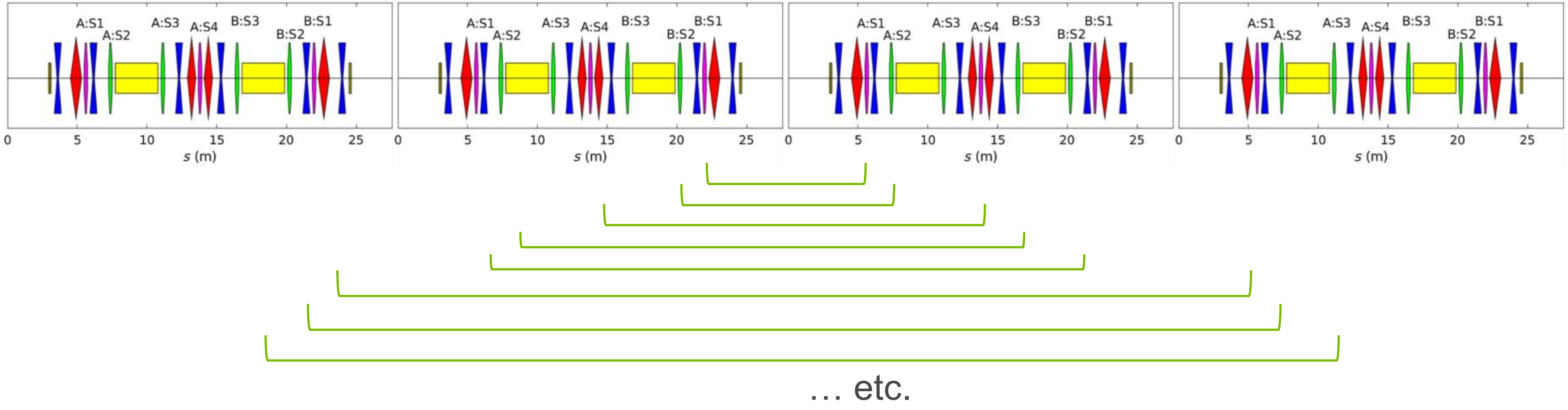


15 steps!



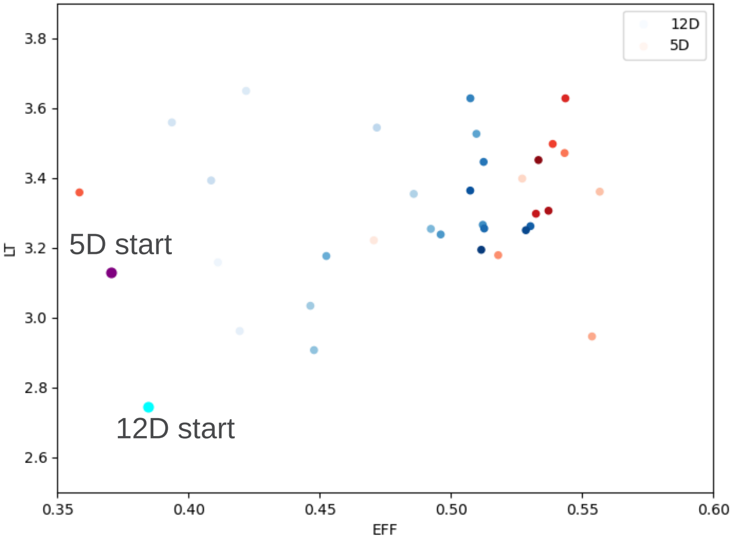
GETTING MORE KNOBS

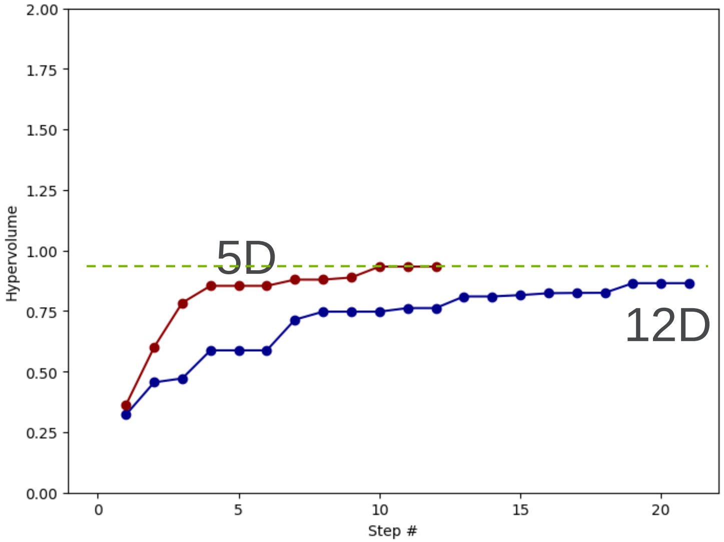
- To get more sextupole knobs, continues to break symmetry with larger supercells
- 14D -> get 12D null space



FINAL DAY OF APS RESULTS

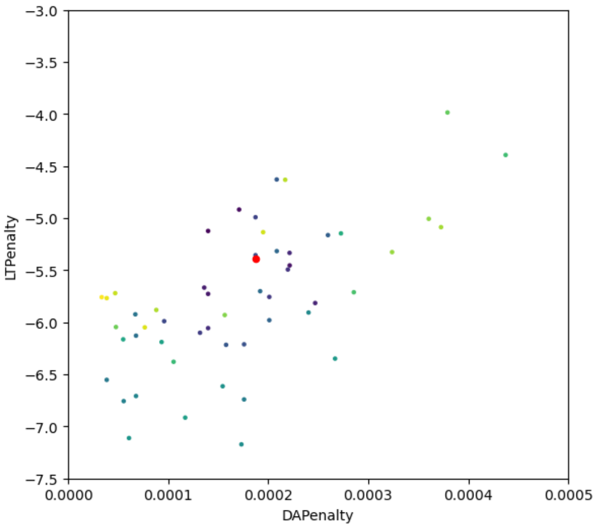
- We followed up on the last shift before APS shutdown to test **5D and 12D on same lattice**
  - Should converge to same performance
  - Observations agree, and 12D is a bit slower as expected

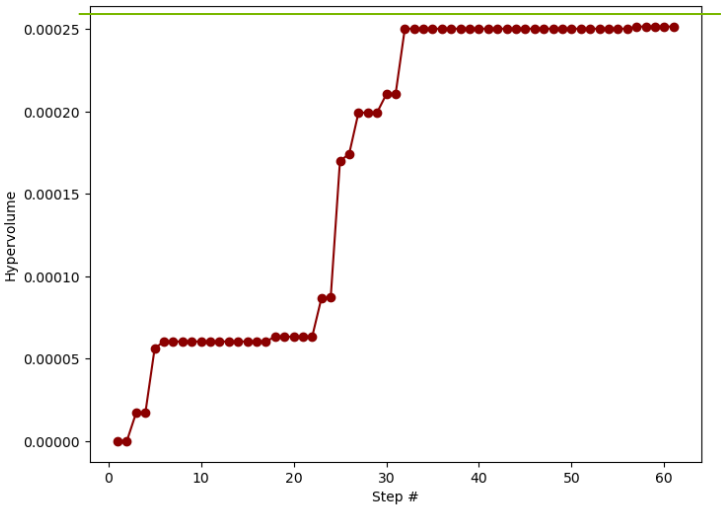




APS-U SIMULATIONS

- MOBO was implemented into APS-U lattice tuning simulation
- Sextupoles only = 12D





- Sextupoles + quad K1 + MB K1 = up to 24D (WIP)

APS-U SIMULATIONS

Several strategies implemented to improve performance (ask for details)

Dynamic reference point

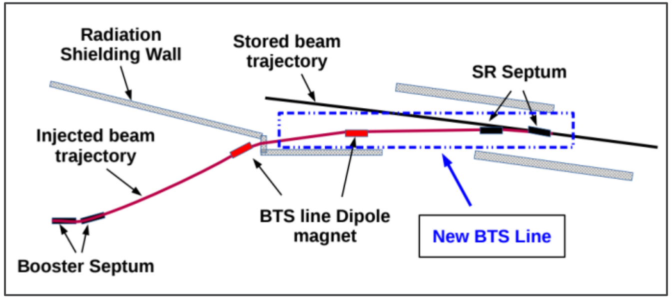
- Move ref point closer to corner of pareto front as simulation progresses (use population quantiles)
- Removes the extreme candidates from consideration, focuses on more relevant region

Surrogate model prior mean

- Instead of fitting only on data, provide prior mean based on simulation-trained surrogate model
- Encodes ‘prior’ knowledge

$$f(\mathbf{x}) \sim \mathcal{GP}(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}'))$$

ML @ APS - MEASUREMENT



Switching topics, now have:

- Have model, don't know parameters
- Want to estimate parameters from data

This case encompasses many standard tasks:

- LOCO and related analyses (i.e. get tilt of magnet, get optics functions)
- Phase space reconstruction (i.e. determine (x,px) at location based on BPM readings, get sigma matrix)

Several applications in APSU can benefit from more efficient parameter 'inference':

- Lattice correction / detection of construction errors
- BTS transfer line measurement

Can ML help?

V. Sajaev, IPAC2015, 556.
R. Lindberg, FLS 2018
APS-U FDR (2019)

BAYESIAN PARAMETER INFERENCE

Idea – adopt a Bayesian approach to **infer parameter posterior distributions**

- Commonly used in other fields (cosmology, HEP)

Fast variants enabled by recent advances:

- Automatic differentiation frameworks for ML (TensorFlow, PyTorch, Jax)
- Probabilistic programming languages (Jax, Pyro)
- New Hamiltonian Monte Carlo samplers (i.e. No U-Turn Sampler)

Together, allow for significantly more efficient estimation of posterior distributions

BAYESIAN BEAM OPTICS

Step 1 – differentiable beam dynamics model

for example, response matrix!

Differentiable = get <observable> + ∂<observable>/∂<parameter>

Need to convert all standard operations (matrix multiply, symplectic integrators, …) to use special functions provided by auto-diff libraries

$$R_{\text{quad}} = \begin{pmatrix} \cos(\omega L) & \frac{\sin(\omega L)}{\omega} & 0 & 0 & 0 & 0 \\ -\omega\sin(\omega L) & \cos(\omega L) & 0 & 0 & 0 & 0 \\ 0 & 0 & \cosh(\omega L) & \frac{\sinh(\omega L)}{\omega} & 0 & 0 \\ 0 & 0 & \omega\sinh(\omega L) & \cosh(\omega L) & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & \frac{L}{\beta_0^2\gamma_0^2} \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix},$$

$$R_{\text{drift}} = \begin{pmatrix} 1 & L & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & L & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & \frac{L}{\beta_0^2\gamma_0^2} \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

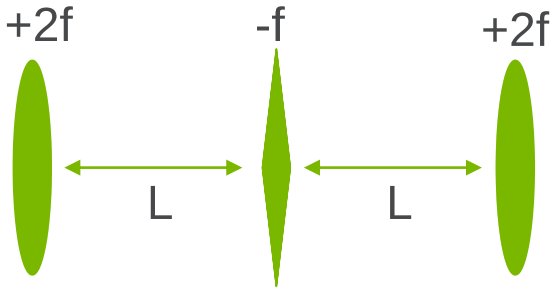
PYRO

+ NumPyro

```
def mat_drift_6d(z, igammabeta2):
    u_matrix = jnp.eye(6)
    u_matrix = u_matrix.at[0,1].set(z)
    u_matrix = u_matrix.at[2,3].set(z)
    u_matrix = u_matrix.at[5,6].set(z*igammabeta2)
    return u_matrix
```

BAYESIAN BEAM OPTICS EXAMPLE

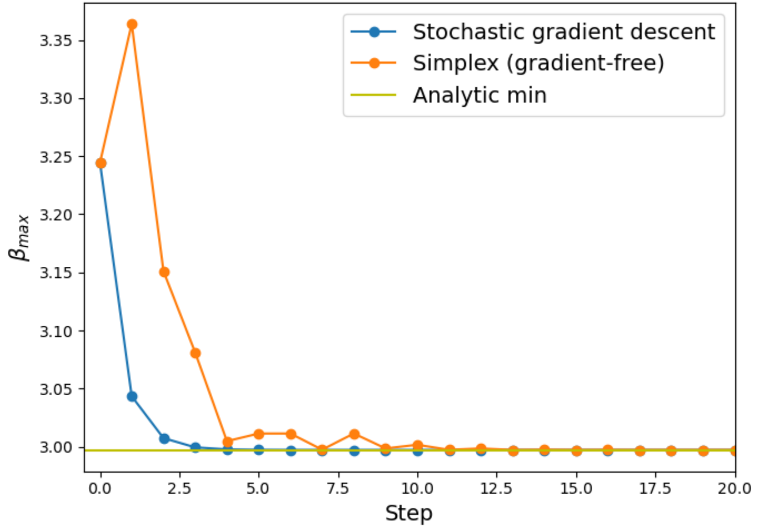
Can use gradient information for fast optics matching.
Example: find min beta phase advance in FODO cell



Get this from autodiff

$$\hat{\beta} = \frac{(1+\sin\frac{\psi_{cell}}{2})L}{\sin\psi_{cell}}$$

$$\frac{d}{d\psi_{cell}}(\hat{\beta}) = \frac{d}{d\psi_{cell}}\frac{L(1+\sin\frac{\psi_{cell}}{2})}{\sin\psi_{cell}} = 0 \quad \rightarrow \quad \psi_{cell} = 76^{\circ}$$



BAYESIAN PARAMETER INFERENCE

Step 2 – set up **efficient posterior estimation**

Disclaimer: not an expert on specific implementations!

Bayes' rule tells us how to use it to perform **inference**, or draw conclusions about latent variables from data, by computing the **posterior distribution**

Posterior of parameters
Want this!

Likelihood

Observations

Prior belief on parameters

$$P(\theta|x) = \frac{P(x|\theta)P(\theta)}{P(x)}$$

'Evidence', very expensive computation

$$P(x) = \int_{\Theta} P(x,\theta)\,\mathrm{d}\theta$$

Model parameters

BAYESIAN PARAMETER INFERENCE

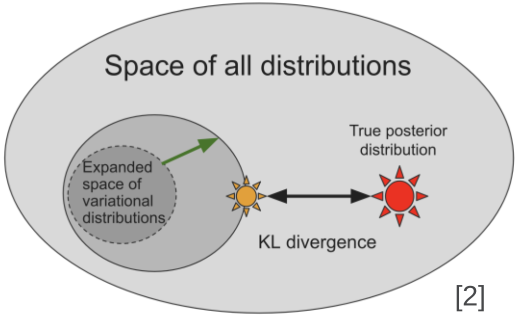
1. arXiv:1206.7051
2. pyro.ai/examples/intro_long.html
3. https://en.wikipedia.org/wiki/Metropolis%E2%80%93Hastings_algorithm#/media/File:Flowchart-of-Metropolis-Hastings-M-H-algorithm-for-the-parameter-estimation-using-the.png

Two main methods:

Variational inference
"a family of techniques for approximating intractable integrals"

Key idea: use a more tractable set of distributions (**variational distributions**) and make them as similar as possible to the desired one

Faster, inherently biased, no convergence guarantees

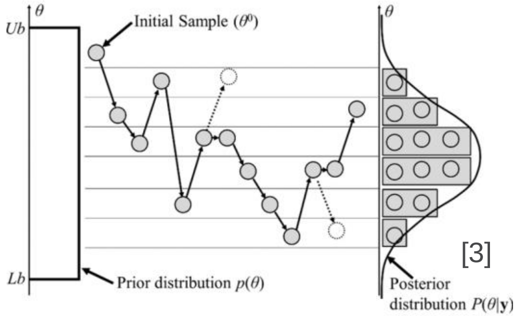


[2]

Markov Chain Monte Carlo
Think of it as jumping around in parameter space with probability of taking a step weighted by the likelihood

With differentiable model, can use an advanced MCMC algorithm – **Hamiltonian Monte Carlo** (HMC)

Slower, but asymptotically exact

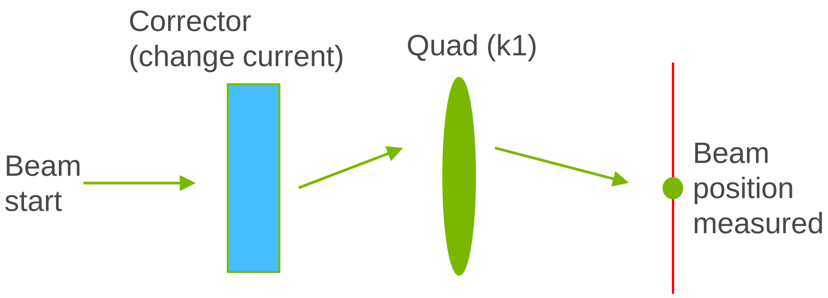


[3]

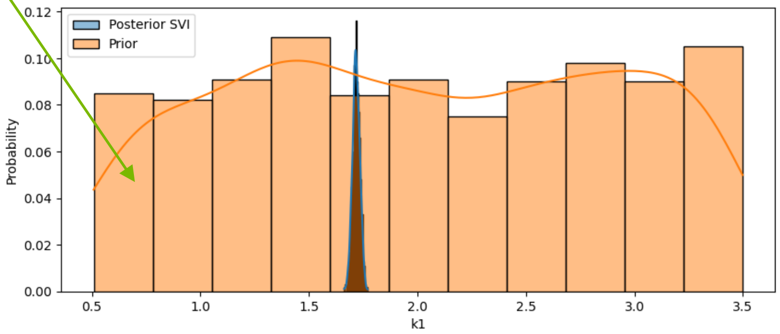
SIMPLE INFERENCE PROBLEM

Goal: determine quadrupole strength *k1* from final beam position data

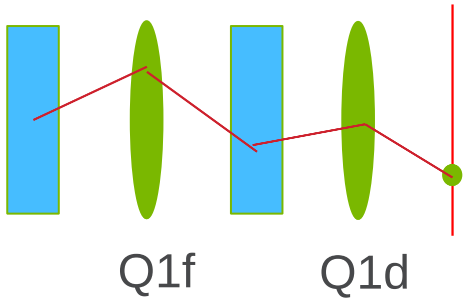
- Model has only 1 latent variable **k1**
- Broad uniform prior (i.e. “its somewhere here”)
- Get correct result!



Sim params: K1= 1.7
noise=100um
10 points (vary corrector)

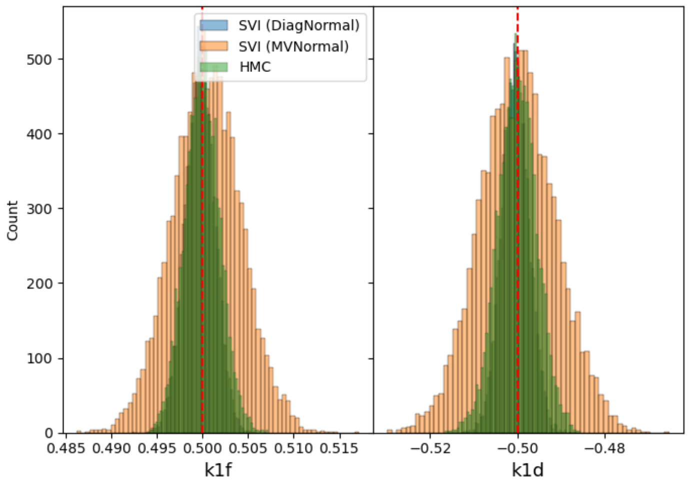


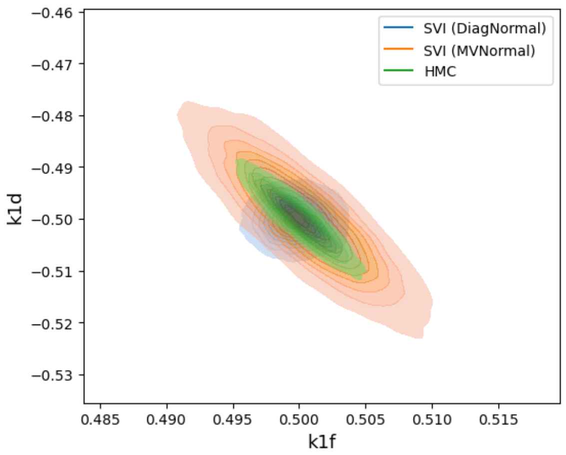
2 QUADS FROM APS LINAC



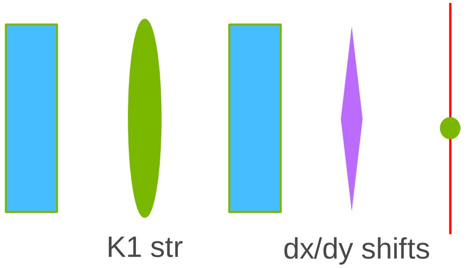
Simulation results: [2 quads + 2 correctors]

- HMC shows expected correlation between parameters
- Simple SVI shows no correlation, needed to use a more complex ‘distribution space’ (multivariate normals)



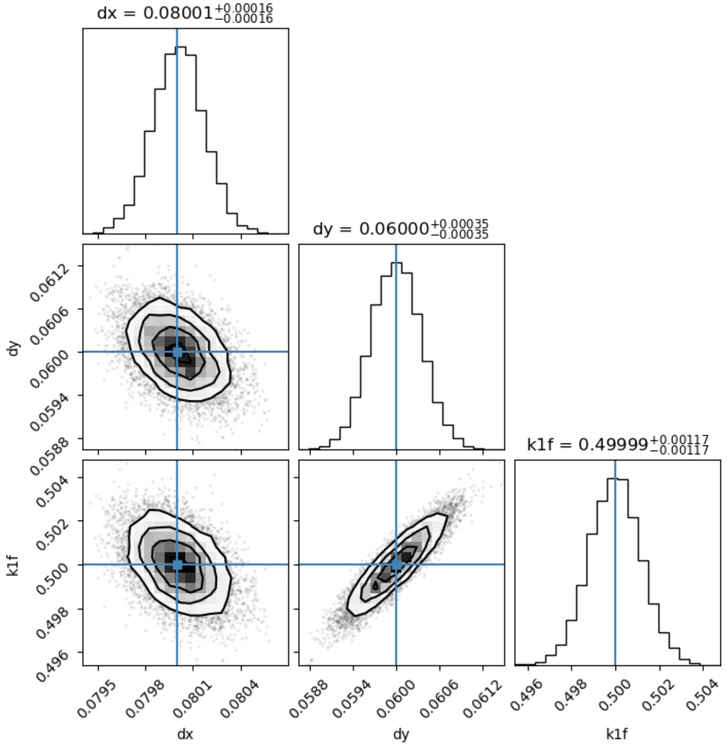


APS LINAC SIMULATIONS



Simulation results: [1 quads +1 sextupole + 2 correctors]

- Sextupole is a nonlinear element, shifted on purpose
- Fitting nonlinear components can be done in LOCO with feed-downs, but we can infer **directly from integrator tracking model**
- HMC works very well (SVI had slight systematic bias)
- Note that sextupole has correlations with quad, consistent with feed-down



APS LINAC SIMULATIONS

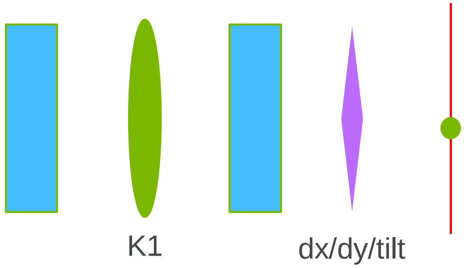
There is no free lunch

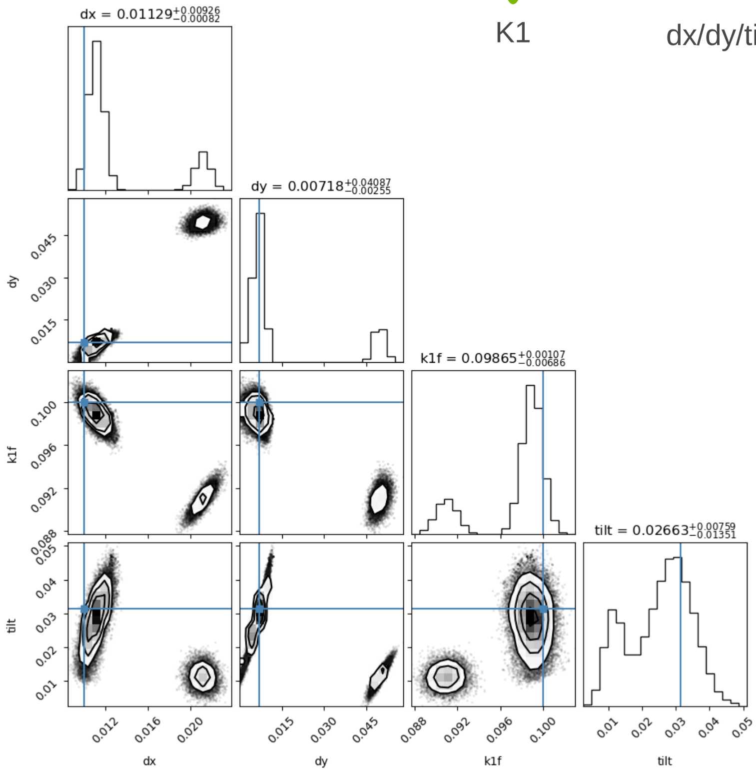
Still need enough data at good locations

Example:

Add sextupole tilt parameter,

get some divergence in fits





INFERENCE SCALING

- Further increasing problem size slows inference but HMC can be parallelized well
- Obtaining good estimates requires better priors (i.e. K1 +- 5%, not full range of magnet)

Is LOCO-scale possible?

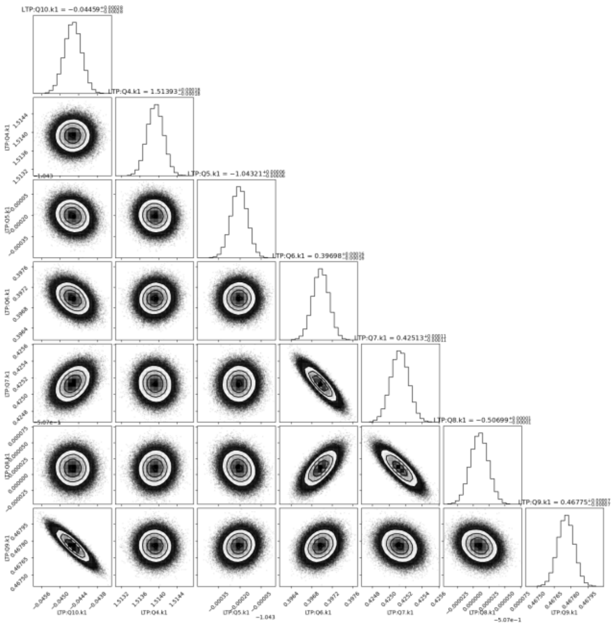
HMC: ~1k parameters reported, slow

SVI: 100k+ reported

Bonus: can speed up measurement by changing many correctors at a time ('fast ORM'). Optimal patterns TBD.

What else can be analyzed?

Beam properties (i.e. sigma matrix), anything with a model



Current focus on performance, and applying to APS BTS transfer line with few dozen params

CONCLUSIONS

Accelerator tuning is a well-studied process, but there are places where **ML can help**

Multi-objective optimization:

No good model, want to tune outputs

- Implemented MOBO algorithm for online tuning
- Obtained faster convergence in APS than previous experimental tests
- Demonstrated feasibility for online applications in full 24D APS-U lattice

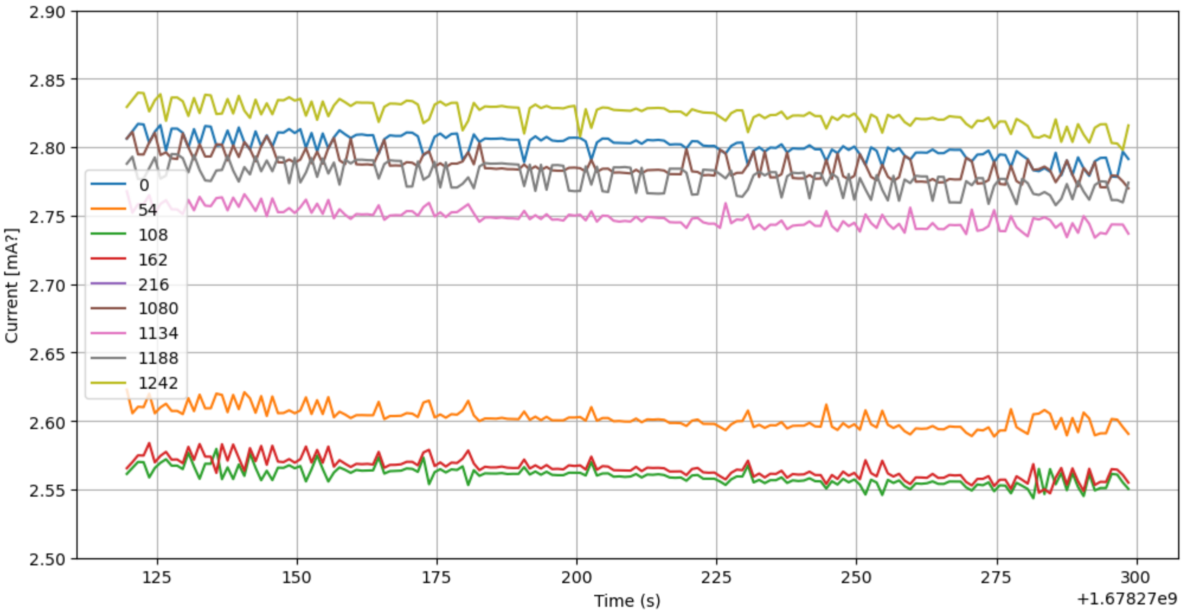
Parameter inference:

Have (complicated) model, want to fit parameters based on data

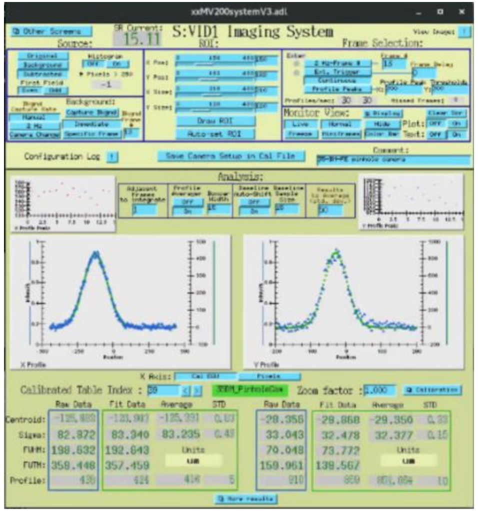
- ML frameworks provide auto-diff capabilities, can make differentiable models
- By using HMC/SVI, get fast Bayesian parameter inference
- Proof of concept for estimating magnet properties – can use for elements not tractable with standard tools

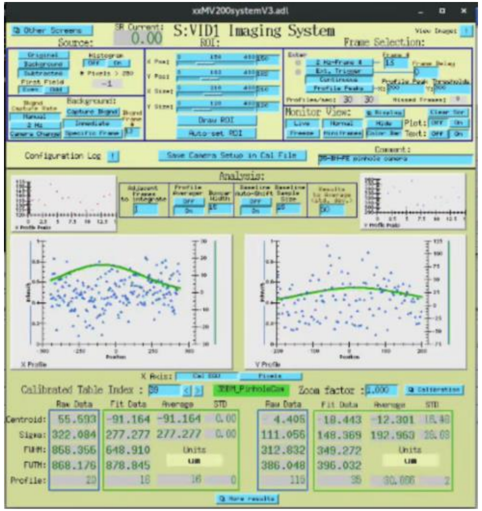
THANK YOU!

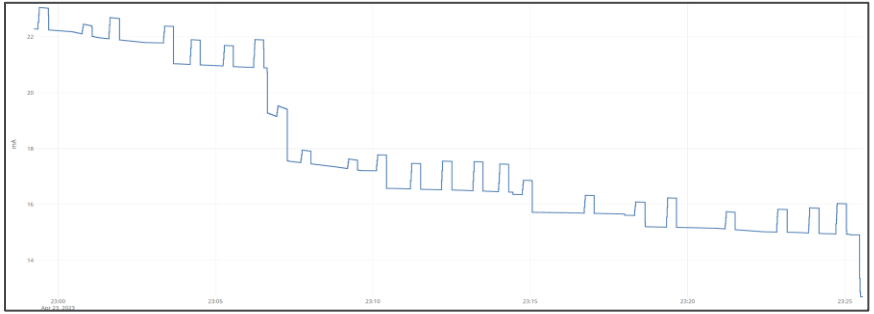
BONUS – BCM RAW DATA



BONUS – RAW SEXTUPOLE INPUTS WITH LARGE RANGE CAUSE MOBO TO CHEAT



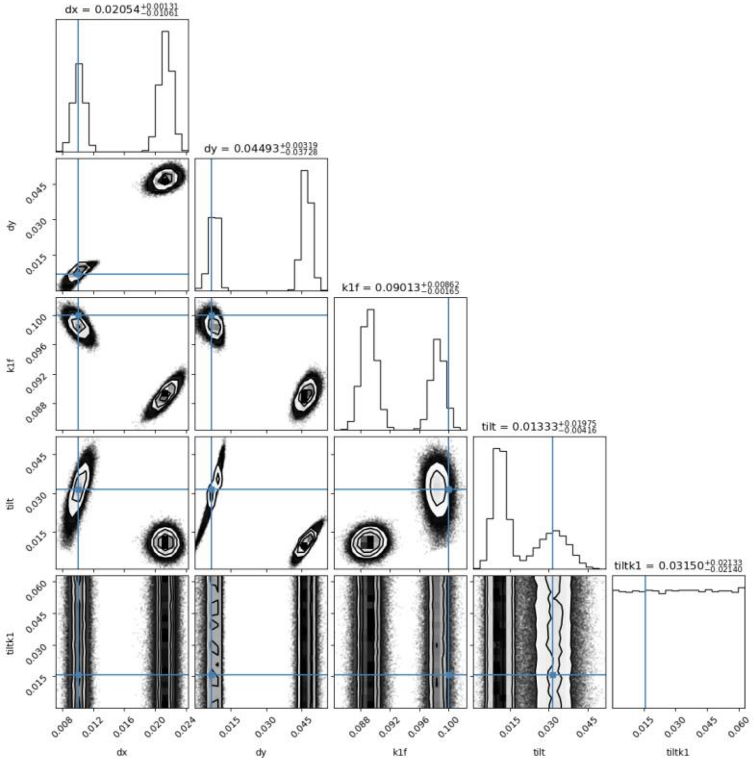




MOBO quickly learned to **cheat lifetime by inducing instabilities**

Bug discovered in knob multiplication (null space factors did not apply, so knobs were currents)

BAD MODEL



$dx = 0.02054^{+0.00131}_{-0.01061}$
$dy = 0.04493^{+0.00319}_{-0.03728}$
$k1f = 0.09013^{+0.00862}_{-0.00165}$
$tilt = 0.01333^{+0.01975}_{-0.00416}$
$tiltk1 = 0.03150^{+0.02133}_{-0.02140}$
dx
dy
k1f
tilt
tiltk1

UNDERCONSTRAINED MODEL